

Master-Theorem

Master-Theorem I

Für $t(n) = a \cdot t(\frac{n}{b}) + g(n) = \sum_{i=0}^{\log_b(n)} a^i \cdot g(\frac{n}{b^i})$

Mit $a > 0, b > 1$ und $g \in \Theta(n^c)$:

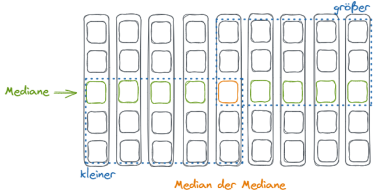
Fall 1	$a < b^c$	$t(n) \in \Theta(n^c)$
Fall 2	$a = b^c$	$t(n) \in \Theta(n^c \log(n))$
Fall 3	$a > b^c$	$t(n) \in \Theta(n^{\frac{\log a}{\log b}})$ bemerke : $\frac{\log a}{\log b} > c$

Master-Theorem II

$T(n) \leq \sum_{i=1}^r T(\alpha_i n) + \mathcal{O}(n)$ für $\sum_{i=1}^r \alpha_i < 1$
 $\Rightarrow T(n) \in \mathcal{O}(n)$

Ultimate-Heapsort (Median in Linearzeit)

- Median aus 5 Elementen ($\frac{n}{5}$ viele Blöcke mit je 6 Vergleichen)
- Median der Mediane
(rekursiv $\Rightarrow T(\frac{n}{5})$)



$T(n) = \frac{6}{5}n + T(\frac{n}{5}) + n + T(\frac{7}{10}n)$

n : Quicksort-Schritte

$\frac{3}{10}$ können durch Median ausgeschlossen werden

bemerke : $\frac{1}{5} + \frac{7}{10} < 1 \Rightarrow$ Master-Theorem II $\Rightarrow \mathcal{O}(n)$

Euklidischer Algo

größter gemeinsamer Teiler **ggT**(m, n)

$\text{ggT}(n \bmod m, m) = \text{ggT}(m, m)$

einfacher Euklid

	99	=	1 · 78	+ 21
	78	=	3 · 21	+ 15
Beispiel ggT(99, 78) :	21	=	1 · 15	+ 6
	15	=	2 · 6	+ 3
	6	=	2 · 3	+ 0

Sobald **Rest = 0** ist der Divisor der ggT (hier 3).

Laufzeit max. $\frac{3}{2} \log_2 m$ Schritte

Erweiterter Euklidischer Algo

Lemma von Bézout : $\text{ggT}(m, n) = am + bn$
(ggT immer als Linearkombination darstellbar)

a	b	q	r	x	y
23	73	3	6	3	-7-7·3 = -4
73	6	2	5	-7	7-2·(-7) = 3
6	5	7	7	7	0-7·7 = -7
5	7	5	0	5	7

$\text{ggT}(23, 73) = 1 = 23 \cdot (-7) + 73 \cdot (-4)$

Einfachen Euklid ausführen. Danach Spalten x, y von unten füllen.

Initial ($x = 0, y = 1$). Danach :

$x_i = y_{i+1}$ und $y_i = x_{i+1} - (q_i \cdot y_{i+1})$

Wenn Multiplikative Inverse benötigt zB : $5 \cdot x \equiv 1 \pmod{13}$

$\Leftrightarrow 5x \bmod 13 = 1 \implies 13a + 5b = 1$ mit erw. Euklid lösen

Restklassenring $\mathbb{Z}/n\mathbb{Z}$

beschreibt "Menge von Mengen".

Einheitsgruppe $(\mathbb{Z}/n\mathbb{Z})^* = \{k + n\mathbb{Z} \mid \text{ggT}(k, n) = 1\}$

Wichtigste Eigenschaften

$(k + n\mathbb{Z}) + (l + n\mathbb{Z}) = k + l + n\mathbb{Z}$ (Addition)

$(k + n\mathbb{Z}) \cdot (l + n\mathbb{Z}) = k \cdot l + n\mathbb{Z}$ (Multiplikation)

$\mathbb{Z}/n\mathbb{Z}$ ist Körper $\Leftrightarrow n$ ist prim

Chinesischer Restsatz

Für Teilerfremde Zahlen m, n :

Abbildung $\varphi : \mathbb{Z}/mn\mathbb{Z} \rightarrow \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$

$x + mn\mathbb{Z} \mapsto (x + m\mathbb{Z}, x + n\mathbb{Z})$

ist Isomorphismus von Ringen.

Folgerung : unendlich viele Primzahlen

Kleiner Satz von Fermat

(Verallgemeinerung von **Satz von Euler** :

$\forall a, n \in \mathbb{N} : \text{ggT}(a, n) = 1 \implies a^{\varphi(n)} \equiv 1 \pmod{n}$

Für Primzahl p und $\forall a \in \mathbb{Z} : a^p \equiv a \pmod{p}$

Falls $p \nmid a$ (a kein Vielfaches von p) : $a^{p-1} \equiv 1 \pmod{p}$

Primzahltest (von n) nach Fermat

Wähle a in $\{1, \dots, n-1\}$ zufällig.

Falls $a^{n-1} \bmod n \neq 1 \pmod{n} \implies n$ KEINE Primzahl

Modulo-Tricks

- Satz von Euler (für Exponenten) & Chin. Restsatz (für Modul)
- Satz von Fermat, wenn $\bmod p \rightarrow a^{p-1}$ auskl.
- mod 3 ist Quersumme $\bmod 3$ (mod in Summe $\sum a_i \cdot 10^i$ ziehen)
- binäre Exponentiation
- $x^d \bmod n \Leftrightarrow x^d \bmod (p-1) \bmod p \wedge x^d \bmod (q-1) \bmod q$
mit p, q prim, $n = pq$
- "−1"-Trick

RSA

Primzahlen p und $q, p < q$. Damit $n = p \cdot q, \varphi(n) = (p-1)(q-1)$

Wähle $e > 0$ mit $\text{ggT}(e, \varphi(n)) = 1$ (Euklidischer Algo)

Berechne $d : e \cdot d \equiv 1 \pmod{\varphi(n)} \quad d < n$

$\Leftrightarrow e \cdot d \bmod \varphi(n) = 1$ bzw. $d = e^{-1} \pmod{\varphi(n)}$

$E(x, (n, e)) = x^e \bmod n \quad (n, e) \in K_{\text{pub}}$

$D(y, (n, d)) = y^d \bmod n \quad (n, d) \in K_{\text{priv}}$

Eulers φ -Funktion

$\varphi(n) = |\{k < n : \text{ggT}(k, n) = 1\}|$

Anzahl ganzer teilerfremde Zahlen unter n .

Für prim : $\varphi(p) = (p-1)$ und $\varphi(p^e) = p^{e-1}(p-1) = p^e - p^{e-1}$

Primzahldichte

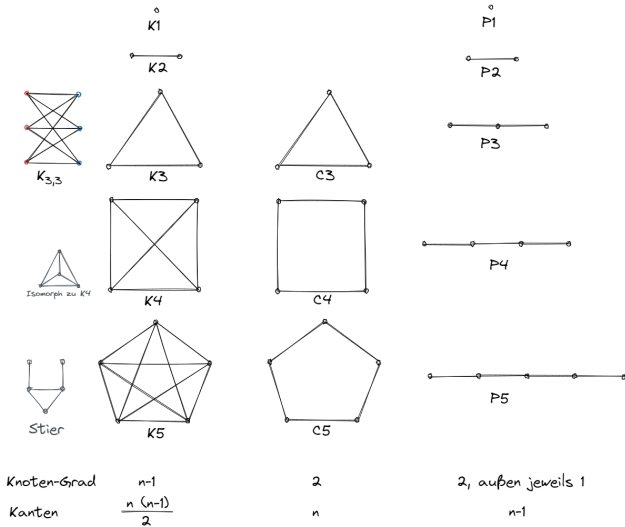
$\pi(n)$ Anzahl Primzahlen bis ink. $n \quad \pi(n) \geq \frac{n}{\log_2 n}$

$\frac{n}{\log_2 n} \leq \pi(n) \leq \frac{(2+\epsilon)n}{\log_2 n}$

Bertrand'sches Postulat $\forall n \geq 1 : \exists p \text{ prim} : n < p \leq 2n$

Ungerichtete Graphen $G = (V, E)$

V, E : Mengen von Knoten, Kanten. $E \subseteq \binom{V}{2}$



bipartit : Knotenmenge V kann in zwei aufgeteilt werden, so dass keine Kante zwei Knoten der gleichen Menge verbindet. Bsp : $K_{3,3}$

d(u) : Grad (degrees)

Summe aller Knotengrade ist gerade (ungerichteter Graph).

Anzahl Knoten mit ungeradem Grad ist gerade!

(Perfect) Matching : So Kanten wählen, dass jeder Knoten mit max. einem anderen Knoten verbunden ist.

Perfect, wenn alle Knoten beteiligt.

Unabhängige Menge : Knoten, die nicht verbunden sind.

Clique

Graph $G, V' \subseteq V$ ist Clique, falls $\forall u, v \in V' : u \neq v \Rightarrow (u, v) \in E$

Satz von Ramsey

$\forall n : \exists N$: Jeder Graph mit N Knoten hat entweder
(eine Clique **oder** unabhängige Menge) der Größe n
Ramsey-Zahl $R(n)$ für kleinsten Graph N

Planar

Isomorpher G. auf Ebene ohne kreuzende Kanten existiert.

G ist planar \Leftrightarrow Untergraph von G enthält keine Unterteilung von

K_5 oder $K_{3,3}$ ~Satz von Kuratowski

Eulerformel

$n - m + f = 2$ n Knoten, m Kanten, f Facetten (**+1 Außen**).

für endliche, zusammenhängende, planare Graphen. $n \geq 1$

Mind. 3 Kanten pro Facette, jede Kante max. 2 Facetten : $3f \leq 2m$

$4f \leq 2m$ bei bipartiten Graphen

Wege und Kreise

Länge von Weg : Kanten!

Euler'scher Weg : Jede Kante einmal in Pfad (**max. 2 Knoten mit ungeradem Grad**)

Euler'scher Kreis : Anfangsknoten = Endknoten (**jeder Knoten gerader Grad**)

Hamilton'scher Weg : Jeder Knoten einmal

Sortieralgos

Da entscheidungsbasiert : mind. Laufzeit von

log
⁡
(
n
!
)
∈
Ω
(
n
log
⁡
n
)

{\displaystyle \log(n!)\in \Omega (n\log n)}

. Also durchquert Baum mit n! Blättern.

Dykstra

Setzte Kosten aller Knoten auf ∞, außer Startknoten (hier 0).

Füge alle Knoten in eine Queue.

Wähle Konten mit kleinstem Wert.

-> Setzte Kosten aller ausgehend verbundenen Knoten auf :

Eigene Kosten + Kosten des Pfades (Wenn niedriger, als die aktuellen)

WIEDERHOLE, bis Queue leer ist.

Beweisbar optimal, Greedy

Bekannte Laufzeiten

Algo	Worst-Case	Average-Case
Quicksort	 O ⁡<!-- ⁡ --> (n 2) {\displaystyle \mathcal {O} (n^{2})} 	 2 n ln ⁡<!-- ⁡ --> n < 1.4 n log ⁡<!-- ⁡ --> n {\displaystyle 2n\ln n<1.4n\log n}
Heapsort	 2 n log ⁡<!-- ⁡ --> n + O ⁡<!-- ⁡ --> (n) {\displaystyle 2n\log n+{\mathcal {O}}(n)} 	 2 n log ⁡<!-- ⁡ --> n + O ⁡<!-- ⁡ --> (n) {\displaystyle 2n\log n+{\mathcal {O}}(n)}
Bottom-up Heapsort	 1.5 n log ⁡<!-- ⁡ --> n + o (n log ⁡<!-- ⁡ --> n) {\displaystyle 1.5n\log n+o(n\log n)} 	 n log ⁡<!-- ⁡ --> n + o (n log ⁡<!-- ⁡ --> n) {\displaystyle n\log n+o(n\log n)}

CYK-Algo

Länge	w1	w2	...
1	T1,1	T2,1	...
2	T2,1	T2,2	
...			

T

i
,
j

=
{
A
∈
V
|
A
⇒

G

∗

,

a

i
...

a

i
+
j
−
1

}

{\displaystyle T_{i,j}=\{A\in V|A\Rightarrow _{G}^{*},a_{i...a_{i+j-1}}\}}

Algo optimale Klammerung

Ähnlich zu CYK.

T

i
,
j

=

min

i
⩽
m
<
j

(

T

i
,
m

+

T

m
+
1
,
j

+

n

i
−
1

⋅

n

m

⋅

n

j

)

{\displaystyle T_{i,j}=\min _{i\leqslant m<j}(T_{i,m}+T_{m+1,j}+n_{i-1}\cdot n_{m}\cdot n_{j})}

Benutzte Technik : Memoization (dyn. Programmieren)

Wachstum

Landau-Symbole

f
∈

O
⁡
(
g
)
:

<

f
wächst langsamer als g

{\displaystyle f\in {\mathcal {O}}(g):<f\;{\text{wächst langsamer als }}g}

f
∈

O
⁡
:

≤

f
wächst nicht (wesentlich) schneller als ...

{\displaystyle f\in {\mathcal {O}}:\leqslant f\;{\text{wächst nicht (wesentlich) schneller als ...}}

f
∈
Θ
:

≍

f
wächst genauso schnell wie ..

{\displaystyle f\in \Theta :~\asymp f\;{\text{wächst genauso schnell wie ..}}

f
∈
Ω
:

≥

f
wächst nicht (wesentlich) langsamer als ..

{\displaystyle f\in \Omega :~\geqslant f\;{\text{wächst nicht (wesentlich) langsamer als ..}}

f
∈
ω
:

>

f
wächst schneller als ..

{\displaystyle f\in \omega :>f\;{\text{wächst schneller als ..}}

Beweis

f
(
n
)
∈

O
⁡
(
b
(
n
)
)
:
∃
c
∃

n

0

∀
(
n
⩾

n

0

)
:

f
(
n
)
⩽
c
⋅
b
(
n
)

{\displaystyle f(n)\in {\mathcal {O}}(b(n)):\exists c\exists n_{0}\,\forall (n\geqslant n_{0})\,:\,f(n)\leqslant c\cdot b(n)}

Bekannte Relationen

log
⁡
(
n
!
)
∈
Ω
(
n
log
⁡
n
)

(
Worst-Case
vergleichsbasiertes Sortieren)

{\displaystyle \log(n!)\in \Omega (n\log n)\,\,\,({\text{Worst-Case vergleichsbasiertes Sortieren)}}

Θ
(
1
)
<
Θ
(
log
log
⁡
n
)
<
Θ
(
(
log
log
⁡
n

)

2

)
<
Θ
(
log
⁡
n
)
<
Θ
(

n

)
<
Θ
(
n

)
<
Θ
(

n

2

)

{\displaystyle \Theta (1)<\Theta (\log \log n)<\Theta ((\log \log n)^{2})<\Theta (\log n)<\Theta ({\sqrt {n}})<\Theta (n)<\Theta (n^{2})}

Θ
(
n
⋅
log
⁡
n
)
<
Θ
(

n

2

)
<
Θ
(

2

n

)
<
Θ
(
n
!
)
<
Θ
(

n

n

)
<
Θ
(

2

n

2

)

{\displaystyle \Theta (n\cdot \log n)<\Theta (n^{2})<\Theta (2^{n})<\Theta (n!)<\Theta (n^{n})<\Theta (2^{n^{2}})}

von n! (für n ≥ 2)

(

n
2

)

n
2

<
n
!
<

n

n

{\displaystyle {\tbinom {n}{2}}{\frac {n}{2}}<n!<n^{n}}

log
⁡
(
n
!
)
∈
Θ
(
n
log
⁡
n
)

{\displaystyle \log(n!)\in \Theta (n\log n)}

e
⋅
(

n
e

)

n

⩽
n
!
⩽
n
⋅
e
⋅
(

n
e

)

n

{\displaystyle e\cdot ({\tfrac {n}{e}})^{n}\leqslant n!\leqslant n\cdot e\cdot ({\tfrac {n}{e}})^{n}}

n
!
≈

2
π
n

⋅
(

n
e

)

n

{\displaystyle n!\approx {\sqrt {2\pi n}}\cdot ({\tfrac {n}{e}})^{n}}

 (Stirling-Formel)

von Binomialkoeffizient (n k) {\displaystyle {\tbinom {n}{k}}}

Maximal bei

(

2
n
n

)

{\displaystyle {\tbinom {2n}{n}}}

 bzw.

(

n
⌊

n
2

⌋

)

=
(

n
⌈

n
2

⌉

)

{\displaystyle {\tbinom {n}{\lfloor {\tfrac {n}{2}}\rfloor }}={\tbinom {n}{\lceil {\tfrac {n}{2}}\rceil }}}

∑

k

(

n
k

)

=

2

n

{\displaystyle \sum _{k}{\tbinom {n}{k}}=2^{n}}

 da alle Möglichkeiten.

(

n
⌊

n
2

⌋

)

=
(

n
⌈

n
2

⌉

)

>

2

n

n

{\displaystyle {\tbinom {n}{\lfloor {\tfrac {n}{2}}\rfloor }}={\tbinom {n}{\lceil {\tfrac {n}{2}}\rceil }}>{\frac {2^{n}}{n}}}

 für

n
⩾
3

{\displaystyle n\geqslant 3}

Durchschnittswert

(

n
k

)

{\displaystyle {\tbinom {n}{k}}}

 ist

2

n

n

{\displaystyle {\tfrac {2^{n}}{n}}}

kgV(n) - Kleinstes gemeinsames Vielfaches

kgV(n) = kgV(2, ..., n)

kgV(5, 8) = 40, da Primfaktorzerlegung 5 = 5, 8 = 2 · 2 · 2.

Alle P-faktoren in ihrer höchsten Anzahl zusammenfassen und aufmultiplizieren.

2

n
−
1

<
kgV
⁡
(
n
)
⩽
n
!

{\displaystyle 2^{n-1}<\mathrm {kgV} (n)\leqslant n!}

2

n

<
kgV
⁡
(
n
)
⩽

4

n
−
1

{\displaystyle 2^{n}<\mathrm {kgV} (n)\leqslant 4^{n-1}}

 für

n
⩾
7

{\displaystyle n\geqslant 7}

m
⋅
(

n
m

)

{\displaystyle m\cdot {\tbinom {n}{m}}}

 teilt kgV(n)

Kombinatorik / Stochastik

Zufallsvariable **X** : Ω ↦ ℝ

X,*Y* sind unabhängig, wenn :

P

(
X
=
x
∧
Y
=
y
)
=
P
⁡
(
X
=
x
)
⋅
P
⁡
(
Y
=
y
)

{\displaystyle \mathbb {P} (X=x\wedge Y=y)=\mathbb {P} (X=x)\cdot \mathbb {P} (Y=y)}

Erwartungswert

E

⁡
(
X
)
=

∑

ω
∈
Ω

X
(
ω
)
⋅
P
⁡
(
ω
)

{\displaystyle \mathbb {E} (X)=\sum _{\omega \in \Omega }X(\omega)\cdot \mathbb {P} (\omega)}

Varianz

V

⁡
(
x
)
=

E

⁡
(
(
X
−

E

⁡
(
X
)

)

2

)
=

E

⁡
(

X

2

)
−

E

⁡
(
X

)

2

{\displaystyle \mathbb {V} (x)=\mathbb {E} ((X-\mathbb {E} (X))^{2})=\mathbb {E} (X^{2})-\mathbb {E} (X)^{2}}

Bedingte Wahrscheinlichkeit

P

(
B
|
A
)
=

P
⁡
(
B
∩
A
)
P
⁡
(
A
)

{\displaystyle \mathbb {P} (B|A)={\frac {\mathbb {P} (B\cap A)}{\mathbb {P} (A)}}}

P

(
A
|
B
)
=

P
⁡
(
B
|
A
)
P
⁡
(
A
)
P
⁡
(
B
)

{\displaystyle \mathbb {P} (A|B)={\frac {\mathbb {P} (B|A)\mathbb {P} (A)}{\mathbb {P} (B)}}}

 Satz von Bayes

Markov-Ungleichung

∀
λ
>
0
:

P
⁡
(
X
≥
λ
⋅

E

⁡
(
X
)
)
⩽

1
λ

{\displaystyle \forall \lambda >0:\,\mathbb {P} (X\geqslant \lambda \cdot \mathbb {E} (X))\leqslant {\frac {1}{\lambda }}}

 für

E

⁡
(
X
)
>
0
,
X
ist
ZV.

{\displaystyle \mathbb {E} (X)>0,X{\text{ ist ZV.}}}

Anzahl Ergebnisse

Ziehe k aus n Optionen :		Zurücklegen	
		Ja	Nein
Reihenfolge	Ja	n^k	
	Nein	$\frac{n!}{(n-k)!} = \binom{n}{k} \cdot k! = n^{\underline{k}}$	

Binomialverteilung

P

(
X
=
k
)
=

(
n
k

)

⋅

p

k

⋅
(
1
−
p

)

n
−
k

{\displaystyle \mathbb {P} (X=k)={\tbinom {n}{k}}\cdot p^{k}\cdot (1-p)^{n-k}}

E

⁡
(
X
)
=
n
⋅
p
,

V

⁡
(
X
)
=
n
⋅
p
⋅
(
1
−
p
)

{\displaystyle \mathbb {E} (X)=n\cdot p,\,\mathbb {V} (X)=n\cdot p\cdot (1-p)}

Geometrische Verteilung (*Wartezeitprobleme*)

P

(
X
=
k
)
=
p
⋅
(
1
−
p

)

k

{\displaystyle \mathbb {P} (X=k)=p\cdot (1-p)^{k}}

 (erfolg im *k*-ten Versuch)

E

⁡
(
X
)
=

1
−
p
p

,

V

⁡
(
X
)
=

1
−
p

p

2

{\displaystyle \mathbb {E} (X)={\frac {1-p}{p}},\,\mathbb {V} (X)={\frac {1-p}{p^{2}}}}

Nice to knows

Isomorphismus

ist ein bijektiver **Homomorphismus** :

struktureerhaltende Abbildung :

ϕ
:
(

M

1

,

o

1

,

e

1

)
↦
(

M

2

,

o

2

,

e

2

)

{\displaystyle \varphi :(M_{1},o_{1},e_{1})\mapsto (M_{2},o_{2},e_{2})}

 mit

ϕ
(
m

o

1

m
′
)
=
ϕ
(
m
)

o

2

ϕ
(
m
′
)

{\displaystyle \varphi (m\,o_{1}\,m')=\varphi (m)\,o_{2}\,\varphi (m')}

Injektiv / Surjektiv

Für *X* ↦ *Y* :

Injektiv (linkseindeutig) : jedes y hat höchstens ein x.

Surjektiv (rechtstotal) : jedes y hat mind. ein x. (Jedes Element in Bildmenge wird getroffen)

Primzahlzertifikat für n

∀ Primzahlen *p* :

n
≡
1

mod
⁡
p
:

∃
a
∈

Z

:

{\displaystyle \forall {\text{Primzahlen }}p:\,n\equiv 1\mod p:\,\exists a\in \mathbb {Z} :}

a

n
−
1

≡
1

mod
⁡
n

{\displaystyle a^{n-1}\equiv 1\mod n}

 und

a

n
−
1

p

≢
1

mod
⁡
n

{\displaystyle a^{\frac {n-1}{p}}\not\equiv 1\mod n}

erste Primzahlen :

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, ...

Reihen

∑

k
=
0

n

q

k

=

1
−

q

n
+
1

1
−
q

{\displaystyle \sum _{k=0}^{n}q^{k}={\frac {1-q^{n+1}}{1-q}}}

 geom. Teil-Reihe

∑

k
=
0

∞

q

k

=

1
1
−
q

{\displaystyle \sum _{k=0}^{\infty }q^{k}={\frac {1}{1-q}}}

 für

|
q
|
<
1

{\displaystyle |q|<1}

 geom. Reihe

∑

k
=
1

∞

k

q

k
−
1

=

1
(
1
−
q

)

2

{\displaystyle \sum _{k=1}^{\infty }kq^{k-1}={\frac {1}{(1-q)^{2}}}}

 für

|
q
|
<
1

{\displaystyle |q|<1}

 geom. Reihe abgeleitet

∑

k
=
1

n

k
=

n
(
n
+
1
)
2

{\displaystyle \sum _{k=1}^{n}k={\frac {n(n+1)}{2}}}

 gaußsche Summenformel

Harmonische Zahl

H

n

=

∑

i
=
1

n

1
i

≈
ln
⁡
(
n
)

{\displaystyle H_{n}=\sum _{i=1}^{n}{\frac {1}{i}}\approx \ln(n)}

ln
⁡
n
⩽

H

n

⩽
ln
⁡
n
+
1

{\displaystyle \ln n\leq H_{n}\leq \ln n+1}

Logarithmus-Regeln

log
⁡
(
x
⋅
y
)
=
log
⁡
x
+
log
⁡
y

{\displaystyle \log(x\cdot y)=\log x+\log y}

log

a

⁡
x
=

log
⁡

b

x

log
⁡

b

a

{\displaystyle \log _{a}x={\frac {\log _{b}x}{\log _{b}a}}}

a

log
⁡
(
b
)

=

b

log
⁡
(
a
)

{\displaystyle a^{\log(a)}=b^{\log(a)}}

Binomialkoeffizienten

Wie viele *k*-elementige Teilmengen existieren von

[
n
]
?

{\displaystyle [n]\,?}

(

n
k

)
=

n
!

k
!
(
n
−
k
)
!

=
(

n
−
1
k

)
+
(

n
−
1
k
−
1

)

{\displaystyle {\tbinom {n}{k}}={\frac {n!}{k!(n-k)!}}={\tbinom {n-1}{k}}+{\tbinom {n-1}{k-1}}}

(

0
0

)
=
(

n
n

)
=
1

{\displaystyle {\tbinom {0}{0}}={\tbinom {n}{n}}=1}

 und

(

n
1

)
=
(

n
n
−
1

)
=
n

{\displaystyle {\tbinom {n}{1}}={\tbinom {n}{n-1}}=n}

(

n
k

)
=
(

n
n
−
k

)

{\displaystyle {\tbinom {n}{k}}={\tbinom {n}{n-k}}}

 (symetrisch)

Satz von Wilson

(
n
−
1
)
!
≡
−
1

mod
⁡
n
↔
n

{\displaystyle (n-1)!\equiv -1\mod n\Leftrightarrow n}

 ist Primzahl

Fibonacci-Zahlen

F

0

=
0,

F

1

=
1,

F

n
+
2

=

F

n
+
1

+

F

n

{\displaystyle F_{0}=0,F_{1}=1,F_{n+2}=F_{n+1}+F_{n}}

F

n

⩽

2

n

⩽

F

2
n

{\displaystyle F_{n}\leqslant 2^{n}\leqslant F_{2n}}

 oder

(

√
2

)

n

⩽

F

n

⩽

2

n

{\displaystyle ({\sqrt {2}})^{n}\leqslant F_{n}\leqslant 2^{n}}

ggT
(

F

m

,

F

n

)
=

F

ggT
(
m
,
n
)

{\displaystyle \mathrm {ggT} (F_{m},F_{n})=F_{\mathrm {ggT} (m,n)}}

Partitionszahlen

n Elemente → k nichtleere Teilmengen aufteilen. Reihenfolge egal.

P
(
n
,
k
)
=
P
(
n
−
1
,
k
−
1
)
+
P
(
n
−
k
,
k
)

{\displaystyle P(n,k)=P(n-1,k-1)+P(n-k,k)}

P

7
,
3

=
4,
da
7
=
1
+
1
+
5
=
1
+
2
+
4
=
1
+
3
+
3
=
2
+
2
+
3

{\displaystyle P_{7,3}=4,da\,7=1+1+5=1+2+4=1+3+3=2+2+3}

Catalanzahlen

C

n

=

1

n
+
1

(

2
n
n

)

=

1

2
n
+
1

(

2
n
+
1

n

)

{\displaystyle C_{n}={\frac {1}{n+1}}{\tbinom {2n}{n}}={\frac {1}{2n+1}}{\tbinom {2n+1}{n}}}

 |

C

1

=
1,

C

2

=
2,

C

3

=
5,

C

4

=
14

{\displaystyle C_{1}=1,C_{2}=2,C_{3}=5,C_{4}=14}

C

n

∼

4

n

n
⋅

√
π
n

{\displaystyle C_{n}\sim {\frac {4^{n}}{n\cdot {\sqrt {\pi n}}}}}

 (durch Stirling)

*C*_{*n*} gibt Anzahl saturierter Binärbäume mit *n* inneren Knoten an (*n* + 1 Blätter)

Dyck-Wörter (Klammerwörter)

a : "(" b : ")"

*D*_{*n*} Menge an Dyck-Wörtern mit Länge 2*n* (also *n* Klammern)

w ∈ *D*_{*n*} wenn

|
w

|

a

=
|
w

|

b

∧
(
∀

w

prefix

aus
w
:

|

w

pref

|

a

⩾
|

w

pref

|

b

)

{\displaystyle |w|_{a}=|w|_{b}\wedge (\forall w_{\mathrm {prefix} {\text{aus}} w}:\,|w_{\mathrm {pref} }|_{a}\geqslant |w_{\mathrm {pref} }|_{b})}

|

D

n

|
=

C

n

{\displaystyle |D_{n}|=C_{n}}

 für

n
⩾
1

{\displaystyle n\geqslant 1}

Induktion

IA, IV & IS.

Für starke Induktion : IV für *m* = 1, 2, ..., *n*

Algebraische Strukturen

Magma : binäre Verknüpfung

∘
:
S
×
S
↦
S

{\displaystyle \circ :S\times S\mapsto S}

Halbgruppe : ∘ assoziativ :

(
x
∘
y
)
∘
z
=
x
∘
(
y
∘
z
)

{\displaystyle (x\circ y)\circ z=x\circ (y\circ z)}

Monoid : (*S*, ∘) : ∃ neutrales Element *e* :

∀
x
∈
S
:
x
∘
e
=
x
=
e
∘